OIPE
JAN 0 5 2004
PATENT & TRADEMARK OFFICE

\# 8

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:
PAU ET AL.

Serial No. **09/712,509**

Confirmation No. **9109**

Filing Date: **November 14, 2000**

For: **METHOD OF VARYING THE BIT RATE OF THE DATA STREAM OF CODED VIDEO PICTURES**

Examiner: **A. RAO**

Art Unit: **2613**

TRANSMITTAL OF CERTIFIED PRIORITY DOCUMENT

RECEIVED
JAN 0 7 2004
Technology Center 2600

COMMISSIONER FOR PATENTS
P.O. BOX 1450
ALEXANDRIA, VA 22313-1450

Sir:

Transmitted herewith is a certified copy of the priority European Application No. 99830713.6.

Respectfully submitted,

Michael W. Taylor

MICHAEL W. TAYLOR
Reg. No. 43,182
Allen, Dyer, Doppelt, Milbrath & Gilchrist, P.A.
255 S. Orange Avenue, Suite 1401
Post Office Box 3791
Orlando, Florida 32802
407-41-2330
407-41-2343 fax
Attorney for Applicant

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: COMMISSIONER FOR PATENTS, P.O. BOX 1450, ALEXANDRIA, VA 22313-1450, on this 2ND day of January, 2004.

Barbara Force

DECLARATION

I, Alberto PELLEGRI, Italian citizen, residing Castelveccana, Via Cesare Battisti, 1E, Italy,hereby declare that I am conversant with the Italian and English languages and that I am the translator of the document attached and certify that to the best of my knowledge and belief the following is a true and correct English translation of the specification and claims contained in the Priority Document No. 99830713.6.

Varese, Italy, 24 December 2003

RECEIVED
JAN 0 7 2004
Technology Center 2600

Alberto PELLEGRI

Europäisches Patentamt | European Patent Office | Office européen des brevets

Bescheinigung | Certificate | Attestation

Die angehefteten Unterlagen stimmen mit der ursprünglich eingereichten Fassung der auf dem nächsten Blatt bezeichneten europäischen Patentanmeldung überein.

The attached documents are exact copies of the European patent application described on the following page, as originally filed.

Les documents fixés à cette attestation sont conformes à la version initialement déposée de la demande de brevet européen spécifiée à la page suivante.

Patentanmeldung Nr. | Patent application No. | Demande de brevet n°

99830713.6

Der Präsident des Europäischen Patentamts;
Im Auftrag

For the President of the European Patent Office

Le Président de l'Office européen des brevets
p.o.

Hatten

I.L.C. HATTEN-HECKMAN

DEN HAAG, DEN
THE HAGUE, 31/05/00
LA HAYE, LE

EPA/EPO/OEB Form 1014 - 02.91

Europäisches Patentamt

European Patent Office

Office européen des brevets

Blatt 2 der Bescheinigung
Sheet 2 of the certificate
Page 2 de l'attestation

Anmeldung Nr.:
Application no.: 99830713.6
Demande n°:

Anmeldetag:
Date of filing: 16/11/99
Date de dépôt:

Anmelder:
Applicant(s):
Demandeur(s):
STMicroelectronics S.r.l.
20041 Agrate Brianza (Milano)
ITALY

Bezeichnung der Erfindung:
Title of the invention:
Titre de l'invention:
MPEG transcoder

In Anspruch genommene Prioriät(en) / Priority(ies) claimed / Priorité(s) revendiquée(s)

Staat: State: Pays:	Tag: Date: Date:	Aktenzeichen: File no. Numéro de dépôt:

Internationale Patentklassifikation:
International Patent classification:
Classification internationale des brevets:
H04N7/26

Am Anmeldetag benannte Vertragstaaten:
Contracting states designated at date of filing: AT/BE/CH/CY/DE/DK/ES/FI/FR/GB/GR/IE/IT/LI/LU/MC/NL/PT/SE
Etats contractants désignés lors du depôt:

Bemerkungen:
Remarks: See for original title page 1 of the description
Remarques:

Italian Text Pursuant to Art. 14.2

"METHOD OF VARYING THE BIT RATE OF THE DATA STREAM OF CODED VIDEO PICTURES"

FIELD OF THE INVENTION

The present invention relates in general to digitized pictures processing and in particular to techniques of bit-rate variation of the data stream of digitized video pictures.

BACKGROUND OF THE INVENTION

The present invention is applicable to processing systems of coded video pictures.

Because of the particular importance of the of the MPEG standard in treating digitized video sequences, reference will be made to an MPEG2 system for concretely illustrating important application of the method of the invention, though the method of the invention remains perfectly usable even in systems for transferring video sequences based on different standards, as established from time to time.

Video coding according to the MPEG2 standard

The MPEG (*Moving Pictures Experts Group*) standard defines a set of algorithms dedicated to the compression of sequences of digitized pictures. These techniques are based on the reduction of the spatial and temporal redundance of the sequence.

Reduction of spatial redundance is achieved by compressing independently the single images, by means of quantization, discrete cosine transform (DCT) and Huffman coding.

The reduction of temporal redundance is obtained using the correlation that exist between successive pictures of a sequence. Approximately, it may be said that each image can be expressed, locally, as a translation of a preceding and/or

successive image of the sequence. To this end, the MPEG standard uses three kinds of pictures, indicated with I (Intra Coded Frame), P (Predicted Frame) and B (Bidirectionally Predicted Frame). The I pictures are coded in a fully independent mode; the P pictures are coded in respect to a preceding I or P picture in the sequence; the B pictures are coded in respect to two pictures, of I or P kind: the preceding one and the following one in the video sequence (re: Fig. 1).

A typical sequence of pictures can be the following one: I B B P B B P B B I B... This is the order in which they will be viewed, but given that any P is coded in respect to the preceding I or P, and any B in respect to the preceding and following I or P, it is necessary that the decoder receive the P pictures before the B pictures, and the I pictures before the P pictures. Therefore the order of transmission of the pictures will be I P B B P B B I B B...

Pictures are elaborated by the coder sequentially, in the indicated order, and successively sent to a decoder which decodes and reorders them, allowing their successive displaying. To code a B picture it is necessary for the coder to keep in a dedicated memory buffer, called "frame memory", the I and P pictures, coded and thereafter decoded, to which current B picture refers, thus requiring an appropriate memory capacity.

One of the most important concepts in coding is motion estimation. Motion estimation is based on the following consideration: a set of pixels of a frame of a picture may be placed in a position of a successive picture obtained by translating the preceding one. Of course, these transpositions of objects may expose parts that were not visible before as well as changes of their shape (e.g. during a zooming and the like).

The family of algorithms suitable to identify and associate these portions of pictures is generally referred to as of "motion estimation". Such an association of pixels is instrumental to calculate a difference picture, thus removing redundant temporal information and making more effective the successive processes of DCT compression, quantization and entropic coding.

Such a method finds in the standard MPEG-2 a typical example. A typical block diagram of a video MPEG-2 coder is depicted in Fig. 2.

Such a system is made of the following functional blocks:

1) Chroma filter block from 4:2:2 to 4:2:0

In this block there is a low pass filter operating on the chrominance component, which allows the substitution of any pixel with the weighed sum of neighbouring pixels placed on the same column and multiplied by appropriate coefficients. This allows a successive subsampling by two, thus obtaining a halved vertical definition of the chrominance.

2) Frame ordinator

This blocks is composed of one or several frame memories outputting the frames in the coding order required by the MPEG standard. For example, if the input sequence is I B B P B B P etc., the output order will be I P B B P B B .... .

- I (Intra coded picture) a frame or a semi-frame containing temporal redundance;
- P (Predicted-picture) is a frame or semi-frame from which the temporal redundance in respect to the preceding I or P (precedingly co/decoded) has been removed;
- B (Biredictionally predicted-picture) is a frame or a semi-frame whose temporal redundance in respect to the preceding I and successive P (or preceding P and successive P) has been removed (in both cases the I and P pictures must be considered as already co/decoded).

3) Estimator

This is the block that removes the temporal redundance from the P and B pictures.

4) DCT

This is the block that implements the cosine-discrete transform according to the MPEG-2 standard.

The I picture and the error pictures P and B are divided in blocks of 8*8 pixels Y, U, V, on which the DCT transform is performed.

5) Quantizer Q

An 8*8 block resulting from the DCT transform is then divided by a quantizing matrix in order to reduce more or less drastically the magnitude of the DCT coefficients. In such a case, the information associated to the highest frequencies, less visible to human sight, tends to be removed. The result is reordered and sent to the successive block.

6) Variable Length Coding (VLC)

The codification words output from the quantizer tend to contain null coefficients in a more or less large number, followed by non null values. The null values preceding the first non null value are counted and the count figure constitutes the first portion of a codification word, the second portion of which represents the non null coefficient.

These pair tend to assume values more probable than others. The most probable ones are coded with relatively short words (composed of 2, 3 or 4 bits) while the least probable are coded with longer words. Statistically, the number of output bits is less than in the case such a criterion is not implemented.

7) Multiplexer and buffer

Data generated by the variable length coder, the quantizing matrices, the motion vectors and other syntactic elements are assembled for constructing the final syntax contemplated by the MPEG-2 standard. The resulting bitstream is stored in a memory buffer, the limit size of which is defined by the MPEG-2 standard requisite that the buffer cannot be overflown. The quantizer block Q attends to the

respect of such a limit, by making more or less drastic the division of the DCT 8*8 blocks depending on how far the system is from the filling limit of such a memory buffer and on the energy of the 8*8 source block taken upstream of the motion estimation and DCT transform steps.

8) Inverse Variable Length Coding (I-VLC)

The variable length coding functions specified above are executed in an inverse order.

9) Inverse Quantization (IQ)

The words output by the I-VLC block are reordered in the 8*8 block structure, which is multiplied by the same quantizing matrix that was used for its preceding coding.

10) Inverse DCT (I-DCT)

The DCT transform function is inverted and applied to the 8*8 block output by the inverse quantization process. This permits to pass from the domain of spatial frequencies to the pixel domain.

11) Motion Compensation and Storage

At the output of the I-DCT may be present either:

- a decoded I frame (or semiframe) that must be stored in a respective memory buffer for removing the temporal redundance in respect thereto from successive P and B pictures or

- a decoded prediction error frame (or semiframe) P or B that must be summed to the information precedingly removed during the motion estimation phase. In case of a P picture, such a resulting sum, stored in dedicated memory buffer is used during the motion estimation process for the successive P pictures and B pictures.

These frame memories are distinct from the frame memories that are used for re-arranging the blocks.

The MPEG2 decoding will be non easily explained by referring to Fig. 3. The first I picture received is decoded by detecting the headers in the bitstream by the HEADER-DETECTION block, a successive inverse VLC decoding, an inverse decoding of the run-level pairs, an inverse quantization, an inverse DCT computation and the successive storing in suitable memory buffers (as described in item 10), and used to calculate the prediction error for decoding the successive P and B pictures.

In video broadcasting, the sequences are transmitted (or are eventually recorded) on a variety of channels and supports, each with its own capacity, speed and cost. Distribution of a film, starting from a master recording, may be made on a DVD (Digital Versatile Disk) or via satellite or cable. The available transmission band may be different from the one allocated during the coding phase of the video sequence, thus raising the problem of re-adapting to the characteristics of new media a bitstream belonging to video pictures, originally coded for a channel with a different bit-rate.

More specifically, this implies the need to modify the bit-rate B1 of a MPEG2 bitstream, expressed in B1 Mbit/s (which is a bandwidth measure of the available channel), generated after a coding of the source sequence, in a bitstream, still coherent to a MPEG2 syntax, with a B2 bit-rate, where B2 is different from B1.

Such a change of bit-rate may be effected in a very simple manner, without using dedicated devices.

Since an encoder and a decoder transform respectively a sequence of photograms into an MPEG2 bitstream and an MPEG2 bitstream into decoded pictures, starting from a bitstream coded with an arbitrary B1 bit-rate, it is always possible to obtain a bitstream with a B2 bit-rate by simply coupling the output of the decoder to the

input of the encoder, after having programmed the latter in order to code with the desired bit-rate B2.

This procedure, which may be defined as an explicit transcoding of a bitstream, requires the following steps:

1. inverse Huffman coding
2. inverse Run-Length coding
3. inverse quantization
4. inverse Discrete Cosine Transform
5. motion compensation

that are carried out in the decoder, while the encoder performs out the following steps:

1. pre-processing
2. motion estimation
3. calculation of the prediction error
4. Discrete Cosine Transform
5. quantization
6. Run-Length coding
7. Huffman coding
8. inverse quantization
9. inverse discrete cosine transform
10. motion-compensation

As it may be easily discerned, such a transcoding process entails a heavy complex computational complexity.

The major computational burden of the above noted sequences dwells in the motion estimation step in the direct/inverse cosine transform steps and in the motion compensation step, whereas quantization, Run-Length coding and Huffman coding are relatively less demanding steps.

There is a need for a method of charging the bit-rate of a data stream of video pictures relatively easier to implement in hardware form and that does not require burdensome calculations.

OBJECT AND SUMMARY OF THE INVENTION

It has been found and is the object of the present invention, a novel method and relative circuits for changing the bit-rate of a bitstream of video pictures with a reduced number of steps and a great simplification of the required hardware resources.

More specifically, the object of the present patent application is a method for producing a bitstream relative to digital video pictures subdividible in a pair of bitstreams of coded data and control bits, respectively, having a bit-rate different from the bit-rate of an input bitstream.

This result is obtained by dividing the input bitstream in a sequence of data and in a sequence of control bits, modifying the sequence of control bits so obtained in function of the different bit-rate to be output producing an output sequence of control bits, decoding the sequence of coded data producing an intermediate sequence of decoded data that is successively quantized with a pre-established step and coded, producing an output sequence of coded data. The output data stream of the desired bit-rate is thence produced by merging the two output sequences. Optionally, it is possible to dequantize the intermediate sequence of decoded data before carrying out the quantization with the pre-established step.

The method of the invention may be easily adapted to pictures coded according to the MPEG2 standard, by making the decoding and coding operations consist respectively in a Huffman decoding followed by a Run-Length decoding, and in a Run-length coding followed by a Huffman coding, respectively.

Preferably, the quantization step is determined by a feed-backward/forward rate control technique or by a feed-backward/forward rate control technique.

A hardware embodiment of this method may comprise a first circuit block separating the input stream in a sequence of coded data and in a sequence of control bits, a second circuit block modifying the sequence of control bits in function of the different bit-rate that is desired, producing an output sequence of control bits, a decoder of said sequence of coded data, producing an intermediate sequence of decoded data, a quantizer with a pre-established step of said intermediate sequence of data, an encoder coupled to the output of the quantizer producing an output sequence of coded data and, finally, a third circuit block merging said two output sequences and outputting a bitstream with the required bit-rate. Optionally, the device may include a dequantizer of the intermediate sequence of decoded data before the quantizer.

BRIEF DESCRIPTION OF THE DRAWINGS

The particular aspects and advantages of the invention will become even more evident through the following description of several embodiments of the invention and by referring to the attached drawings, wherein:

Figure 1 is a diagram comparing the known transcoding technique with the method of the invention;

Figure 2 is a block diagram of an MPEG2 encoder;

Figure 3 is a block diagram of an MPEG2 decoder;

Figure 4 depicts an architecture implementing the method of the invention;

Figure 4 bis shows a preferred embodiment of the quantization chain;

Figure 5 shows a possible architecture of the invention including a control of the feed-backward compression rate;

Figure 6 shows an architecture of the invention including a control of the feed-forward compression rate.

DESCRIPTION OF SEVERAL EMBODIMENTS OF THE INVENTION

The bit-rate is determined by the number of bits dedicated to the coding of the quantized DCT coefficients of the prediction error. By operating on the quantization parameter it is possible to increase or decrease the bit-rate: the selection of this parameter is made by the control module of the compression rate.

The portions of the bitstream that do not influence the reduction of the bit-rate are simply copied rather than processed. This occurs for the majority of the data relative to the syntax and, above all, for the motion vectors, because the motion field depends solely on the calculations carried out on the source sequence.

Since the goal is to obtain a reduction of the bit-rate, by referring to the MPEG2 standard the coding process is equivalently represented by the VLC inverse coding and by the Run-Length decoding.

It is not necessary to carry out a discrete inverse cosine transform and a motion compensation, because the control of the bit-rate occurs in the frequency domain, with a net saving of computational work. Once the quantized DCT coefficients are extracted, they are requantized with a new quantization step, which is typically different from the quantization step found in the input (source) bitstream, and the Run-length and Huffman coding are finally performed.

Optionally, the requantization operation may be preceded by a de-quantization of the DCT coefficients in case it is desired to operate with non quantized values.

Substantially, the computational savings derive from eliminating the motion compensation step, the motion estimation step, the direct and inverse discrete cosine transform steps.

A comparison of the method of the invention and a normal transcoding technique is illustrated in Figure 1. With the method of the invention it is possible to obtain directly an output data stream at the desired bit-rate of B2 Mbit/s, through the block MPEG2_TRANSCODER that implements the method of the invention. This block processes an input data stream having a bit-rate of B1 Mbit/s, as

produced by an MPEG2_ENCODER that codes according to the MPEG2 standard the pixels of a SOURCE_SEQUENCE.

Indeed, the same result may be obtained by connecting in cascade an MPEG2_DECODER and an MPEG2_ENCODER. However, as shown in Figure 1, such a trivial approach implies the generation of a sequence of decoded pictures DECODED_SEQUENCE, with the associated costs in terms of processing complexity and time required.

A basic diagram of a hardware embodiment of the method of the invention is shown in Figure 4. The INPUT_BISTREAM is fed to a circuit block that separates the headers of the sequence and of the GOP (Group Of Pixels), which are successively adapted to the desired bit-rate and sent to a multiplexer that produces the output bitstream.

The Picture_Header_Detection block detects the presence of the control bits of a picture of the video sequence, separating them from the data bits that are sent to a Requantization_Block. This block carries out, after the Huffman and Run-length decoding process, a dequantization and a quantization with a pre-established step, followed by a Run-Length coding and a Huffman coding. The step of quantization step is determined by the Rate_Control and BUP blocks, that store the number of bits produced.

A preferred embodiment of the Requantization_Block is shown in Fig. 4 bis. In the figure can be observed the cascade of blocks performing the above mentioned Huffman decoding (IVLC), Run-Length decoding (IRL) and dequantization (IQ), followed by a quantization (Q), with a step determined by the Quantization_Parameter, and said Run-Length (RL) and Huffman (VLC) coding.

The compression rate may be regulated by means of dedicated multiplexers. Essentially, there are two choises:

1) feed-backward rate control

2) feed-backward/forward (with pre-analysis) hybrid rate control.

The MPEG2, Test Model 5, provides a detailed description of the above mentioned well known techniques for regulating the compression rate. Other implementations of the above noted control techniques are described in the patent applications No.: EP97830591.0, EP98830599.1 and EP99830560.1, in the name of the same applicant.

The feed-backward technique of rate control is susceptible of a hardware embodiment, as the one depicted in Figure 5.

Referring to this figure it may be stated that:

- through the input gate [1] the bitstream is transferred to the [A] block;
- the first parsing takes place in [A]: if data relative to the syntax of the sequence and of the GOP are detected, they are conveyed through [2], toward [B], which extracts certain format data, and the block [L] synchronizes the output thereof;
- when [A] detects the start data of the picture, [C] takes control until a new header of the sequence/GOP/picture is detected;
- when [A] starts to transmit picture data, [C] conveys them through [D] toward the requantization chain [E-J]. In this phase, [K] provides the quantization parameter, and through [7] collects information on the past performance of the coding process;
- [L] reconstructs the bitstream starting from the contribution of [3] (syntax of the sequence and of the GOP), of [5] (motion vectors) and of [6] (DCT coefficients).

The alternative technique of rate control may be implemented by the architecture of Figure 6, wherein:

- through the input gate [1] the bitstream is conveyed to the [A] block;
- [A] carries out a first analysis of the bitstream (parsing): if data relative to the syntax of the sequence and of the GOP are detected, they are conveyed through [2], toward [B] which extracts certain format data, and the block [P] synchronizes the output thereof;
- when [A] detects the start data of a picture, [C] takes control until a new header of the sequence/GOP/picture is detected;
- [C] controls the timing between the pre-analysis and the recoding: when [A] starts to transmit picture data, [C] copies the data in the buffer [E] and simultaneously conveys such data through [6] towards the requantization chain [G-L]. In this phase, [O] provides the quantization parameter during the preanalysis phase, and [M] sends the collected data to [N] which constructs the Bit Usage Profile of the picture;
- thereafter, [C] switches its output from [D] towards [F] so that the original data are read again by [E] and retransmitted to the requantization chain. [O] supplies the quantization parameters of the final coding using the data collected by [N] during the preanalysis. [M] outputs data towards [13] and sends the requantized and coded coefficients to [P]. [C] inserts properly the data output by [M] with the original motion vectors stored in [E] during the pre-analysis phase;
- [P] reconstructs the bitstream starting from the contributions of [3] (syntax of sequence and GOP), of [8] (motion vectors) and of [13] (DCT coefficients).

In order to further illustrate the functioning of the device of Figure 6, reference may be made to the following representation, in a C pseudo-code of the procedures performed by the main circuit blocks.

It is useful to list several functions that are commonly used to access the bitstream:

ShowBitsS(N)

ShowBitsC(N)

show, without moving from the actual position, the next N bits of the bitstream. The first function reads from the input (1), the second function reads from the picture memory (LOCAL_MEMORY).

GetBitsS(N)

GetBitsC(N).

show the next N bits of the input bitstream and of the LOCAL_MEMORY, respectively. The cursor shifts by N positions.

PutBitsD(bits)

PutBitsC(bits)

write the past bits by argument in the output bitstream (14) and in the LOCAL_MEMORY, respectively.

MoveBitsSD(N)

MoveBitsSC(N)

MoveBitsCD(N)

are combinations of GetBits*() e PutBits*() functions which allow to read and move N bits from the input to the output, from the input to the LOCAL_MEMORY and from the LOCAL_MEMORY to the output, respectively.

Block [A]+[B]

The VideoSequence() procedure recognizes the header sections of the sequence and of GOP, and outputs them; and when the start code of the picture is detected passes the control to [C]

```
VideoSequence()
{
  while(NextStartCode() != SEQUENCE_END_CODE)
  {
   if (SEQUENCE_START_CODE || GOP_START_CODE)
   {
    /* copy input data to output */
   }
   else if(PICTURE_START_CODE)
   {
    /* [C] */
   }
  }
}
```

Block [C]

```
TranscodePictureData()
{
```

```
 PicturePreanalysis()
 PictureReshape()
}
```

In the detail:

```
PicturePreanalysis()
{
 for(i=0; i<= MACROBLOCK_COUNT; i++)
 {
  MoveBitsSC(MOTION_VECTORS)
  mQuant = PreanalysisMQuant() /* rate control [O] */
  Requantize(DCTMatrix, mQuant)
  BUP[i] = BitCount(DCTMatrix)
 }
}
```

The letter procedure carries out the parsing of the picture header, then it begins the macroblock loop until the end of the data section of the picture is reached. The bitstream access functions used in this case are the ReadBits(), PutBitsC() and, above all, MoveBitsSC(). This implies that, while the bitstream is "consumed" for performing the preanalysis, the read bits are saved in the LOCAL_MEMORY.

The Requantizer() function is self-explanatory, whereas the BitCount() performs the Run-Length and Huffman coding without writing the result but counting only the bits produced; such a number is written in the BUP.

At the end of this procedure the situation is as follows:

- the pointer to the source bitstream is positioned at the end of the data section of the picture;
- the pointer to the LOCAL_MEMORY is positioned at the start of the LOCAL_MEMORY itself, which contains a copy of the section of data of the picture just read from the source bitstream;
- in the output file none of the bits relative to the picture has been written yet.

```
Picture Reshape()
{
  for(i=0; i<= MACROBLOCK_COUNT; i++)
  {
    MoveBitsCD(MOTION_VECTORS)
    mQuant = ReshapeMQuant() /* rate control */
    Requantize(DCTMatrix, mQuant)
    PutBitsD(DCTMatrix)
  }
}
```

This routine is very similar to a Picture Preanalysis(), apart from the fact that, the functions to access the data become ReadBits(), PutBitsD() and MoveBitsCD().

At the end of this procedure the situation is as follows:

- the pointer to the source bitstream is set at the end of the data section of the picture;

- the pointer to the LOCAL_MEMORY is set at the end of the LOCAL_MEMORY itself, which will be successively emptied in order to receive the data belonging to the successive picture;

- the transcoded data of the picture have been written in the output file.

Block [O]

```
PreanalysisMQuant()

{

 /* returns previous picture's mean mQuant */

}
```

The pre-analysis quantization parameter mQuant is the mean one of the coding of the last homologous picture (i.e.: of the same type I, P or B).

```
ReshapeMQuant()

{

 LocalError = BITS_PRODUCED-BUP

 IntegralError += LocalError

 mQuant = PREANALYSIS_MQUANT + LocalError*PropCoeff +
IntegralError*IntCoeff

}
```

At each coding step, the RATE_CONTROL block measures the offset from the ideal profile (LocalError) and calculates also the integral error. The mQuant value is then obtained by applying to the PI controller the coefficients PropCoeff e IntCoeff.

CLAIMS

1. A method of producing an output bitstream of coded digital video data, with a bit-rate different from the bit-rate of an input bitstream, which comprises the steps of

dividing said input bitstream into a sequence of coded data and a sequence of control bits;

modifying said sequence of control bits in function of the different bit-rate of the output bitstream that is desired, producing an output sequence of control bits;

decoding said sequence of coded data producing an intermediate sequence of data;

quantizing with a pre-established step and coding said intermediate sequence of data producing an output sequence of coded data;

merging said output sequences producing said output bitstream with the desired bit-rate.

2. The method of claim 1, in which said sequence of intermediate data is dequantized before being quantized with said pre-established step.

3. The method according to one of the claims 1 or 2, in which said bitstreams are of MPEG coded digital video pictures, and said decoding and coding steps respectively consist in

performing a Huffman decoding followed by a Run-Length decoding, and

performing a Run-Length coding followed by a Huffman coding.

4. The method of claim 3, wherein said pre-established quantization step is determined by a feed-backward rate control technique.

5. The method of claim 3, wherein said pre-established quantization step is determined by a feed-backward/forward hybrid rate control technique.

6. A device for producing a bitstream of coded digital video data with a bit-rate different from the bit-rate of an input bitstream of coded digital video data comprising

a first circuit block separating said input bitstream in a sequence of coded data and in a sequence of control bits;

a second circuit block fed with said sequence of control bits and outputting a modified sequence of control bits in function of the desired different bit rate;

a decoder of said sequence of coded data, producing an intermediate sequence of data;

a quantizer with a pre-established step of said intermediate sequence of data

an encoder in cascade of the output of said quantizer producing an output sequence of coded data;

a third circuit block merging said output sequence of coded data and said modified sequence of control bits producing said output bitstream with the desired bit-rate.

7. The device of claim 6 comprising a dequantizer of said intermediate sequence of data before said quantizer.

8. The device according to claim 6 or 7 wherein said bitstreams are of MPEG coded digital video data and said decoder and said encoder consist respectively of

a Huffman decoder followed by a Run-Length decoder, and

a Run-length coder followed by a Huffman coder.

9. The device according to anyone of the claims from 6 to 8, wherein said quantization step of said quantizer is set by a bit rate control block coupled to said encoder, and

said third circuit block comprising at least a multiplexer functionally coupled the outputs of said first circuit block, of said second circuit block and of said encoder.

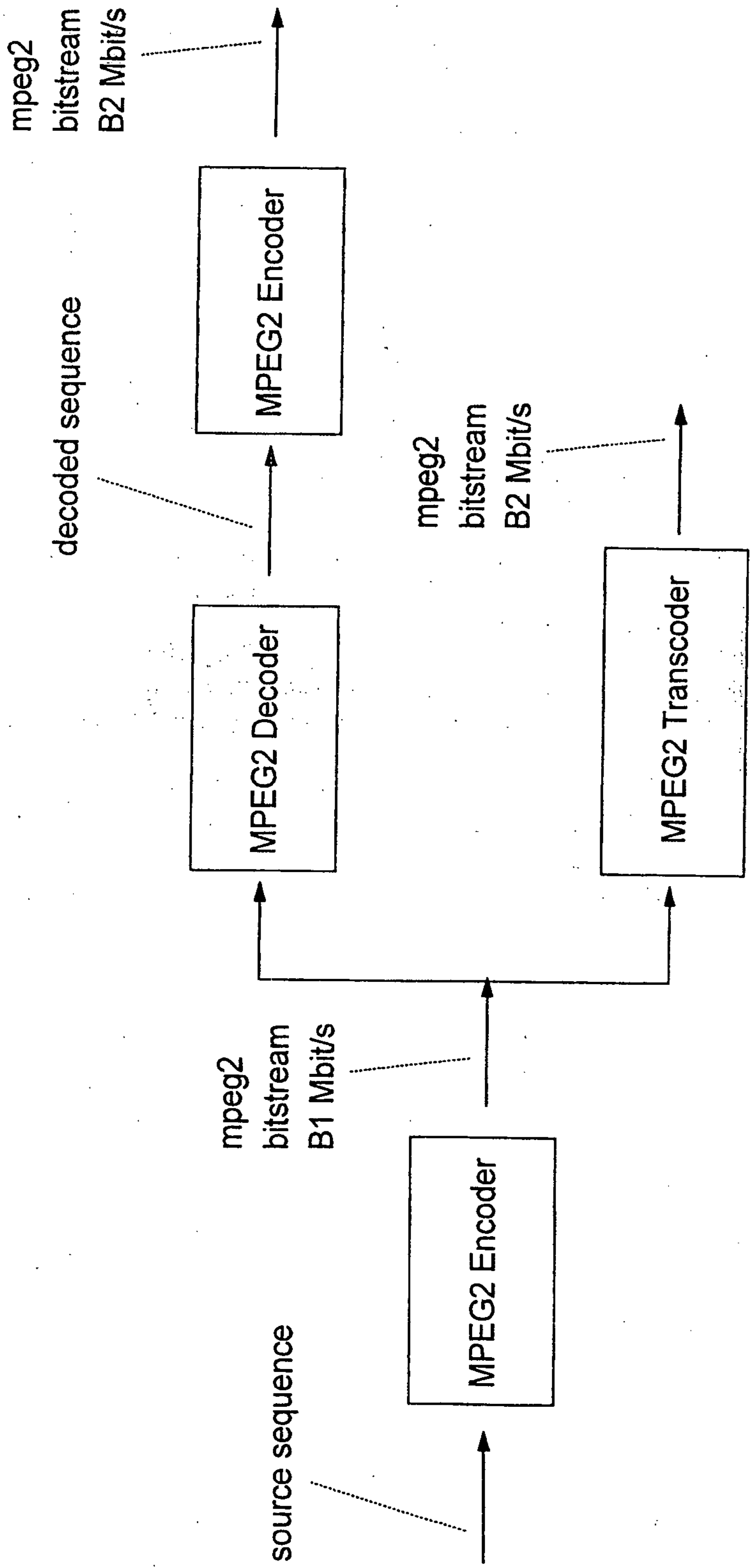
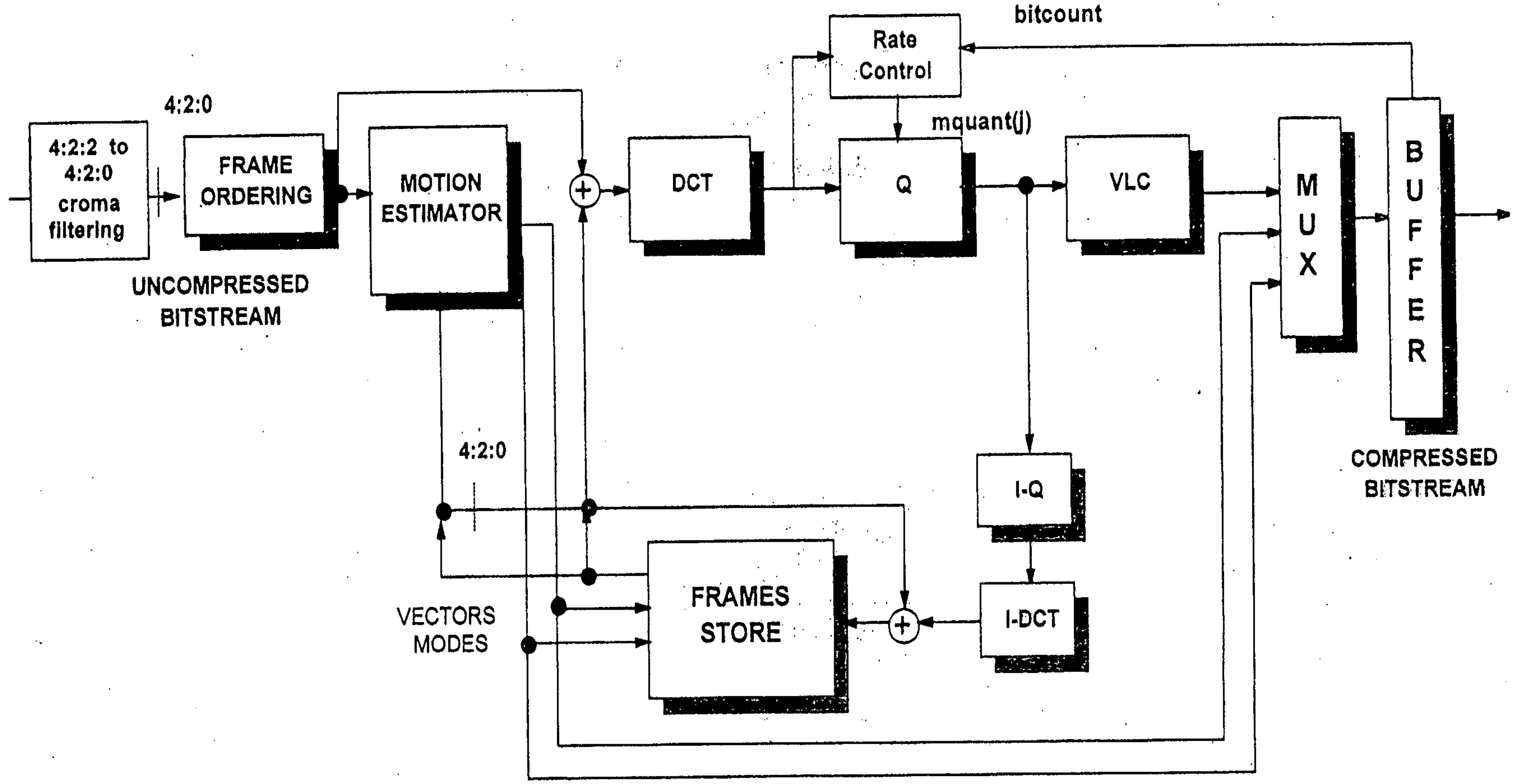

source sequence
MPEG2 Encoder
mpeg2
bitstream
B1 Mbit/s
MPEG2 Decoder
decoded sequence
MPEG2 Encoder
mpeg2
bitstream
B2 Mbit/s
MPEG2 Transcoder
mpeg2
bitstream
B2 Mbit/s

FIG. 1



4:2:2 to 4:2:0 croma filtering
4:2:0
FRAME ORDERING
UNCOMPRESSED BITSTREAM
MOTION ESTIMATOR
DCT
Rate Control
bitcount
mquant(j)
Q
VLC
MUX
BUFFER
COMPRESSED BITSTREAM
4:2:0
I-Q
I-DCT
FRAMES STORE
VECTORS MODES

FIG. 2

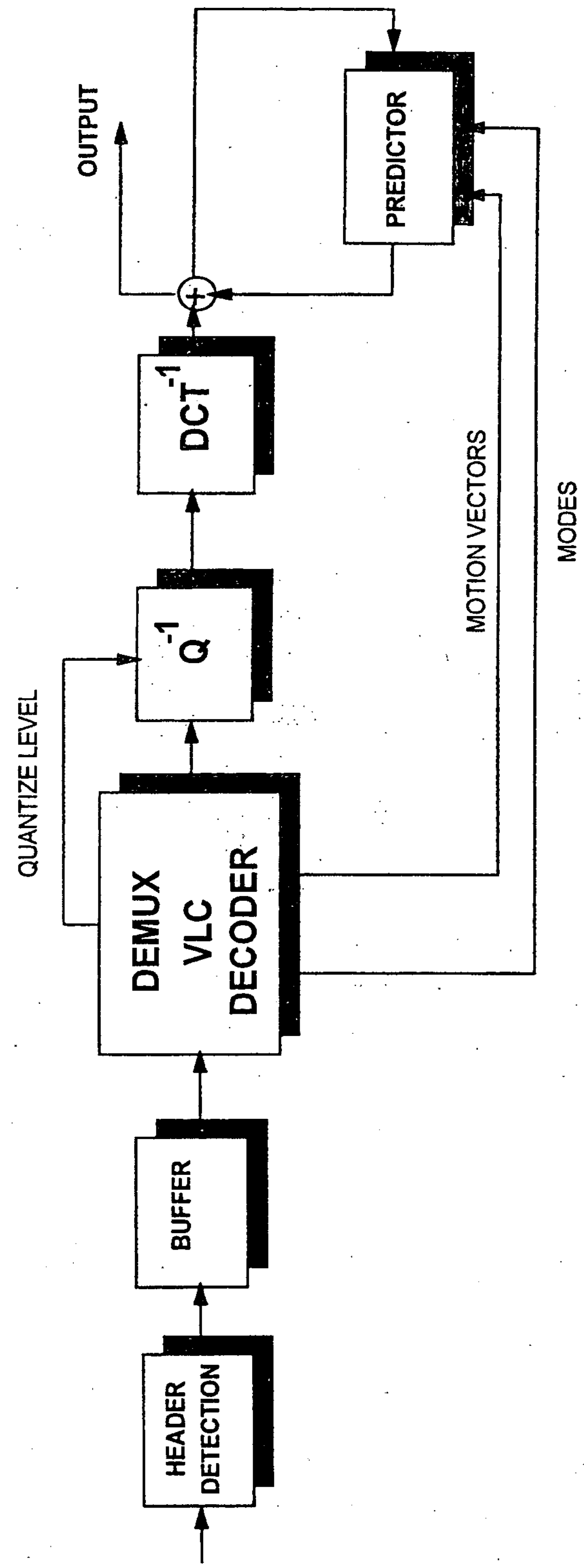

HEADER DETECTION
BUFFER
DEMUX VLC DECODER
QUANTIZE LEVEL
Q^{-1}
DCT^{-1}
OUTPUT
PREDICTOR
MOTION VECTORS
MODES

FIG. 3

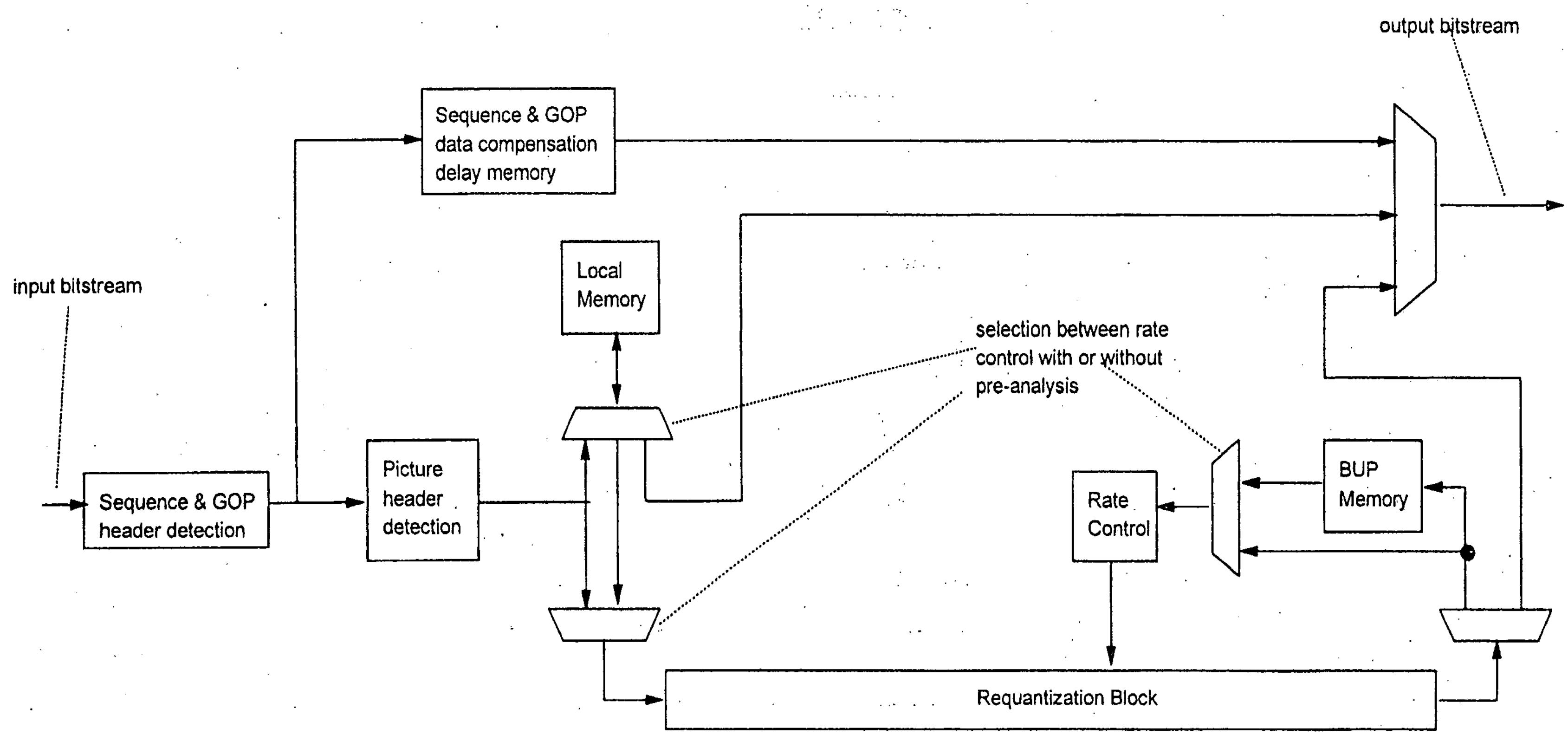

output bitstream
Sequence & GOP data compensation delay memory
input bitstream
Local Memory
selection between rate control with or without pre-analysis
Sequence & GOP header detection
Picture header detection
BUP Memory
Rate Control
Requantization Block

FIG. 4

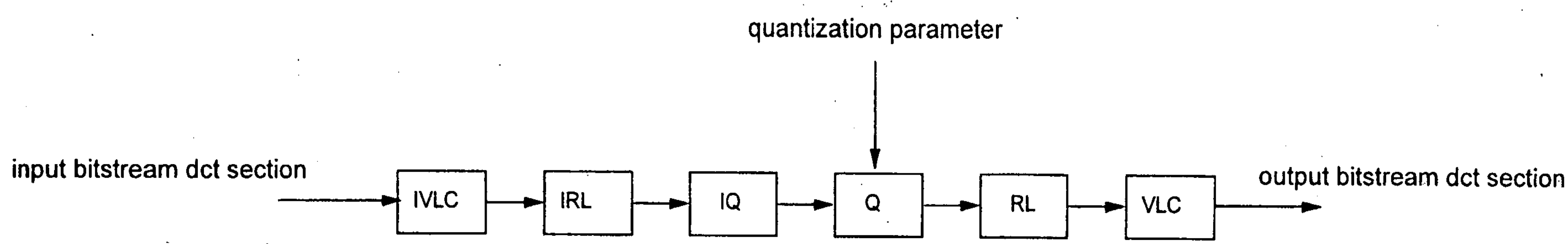
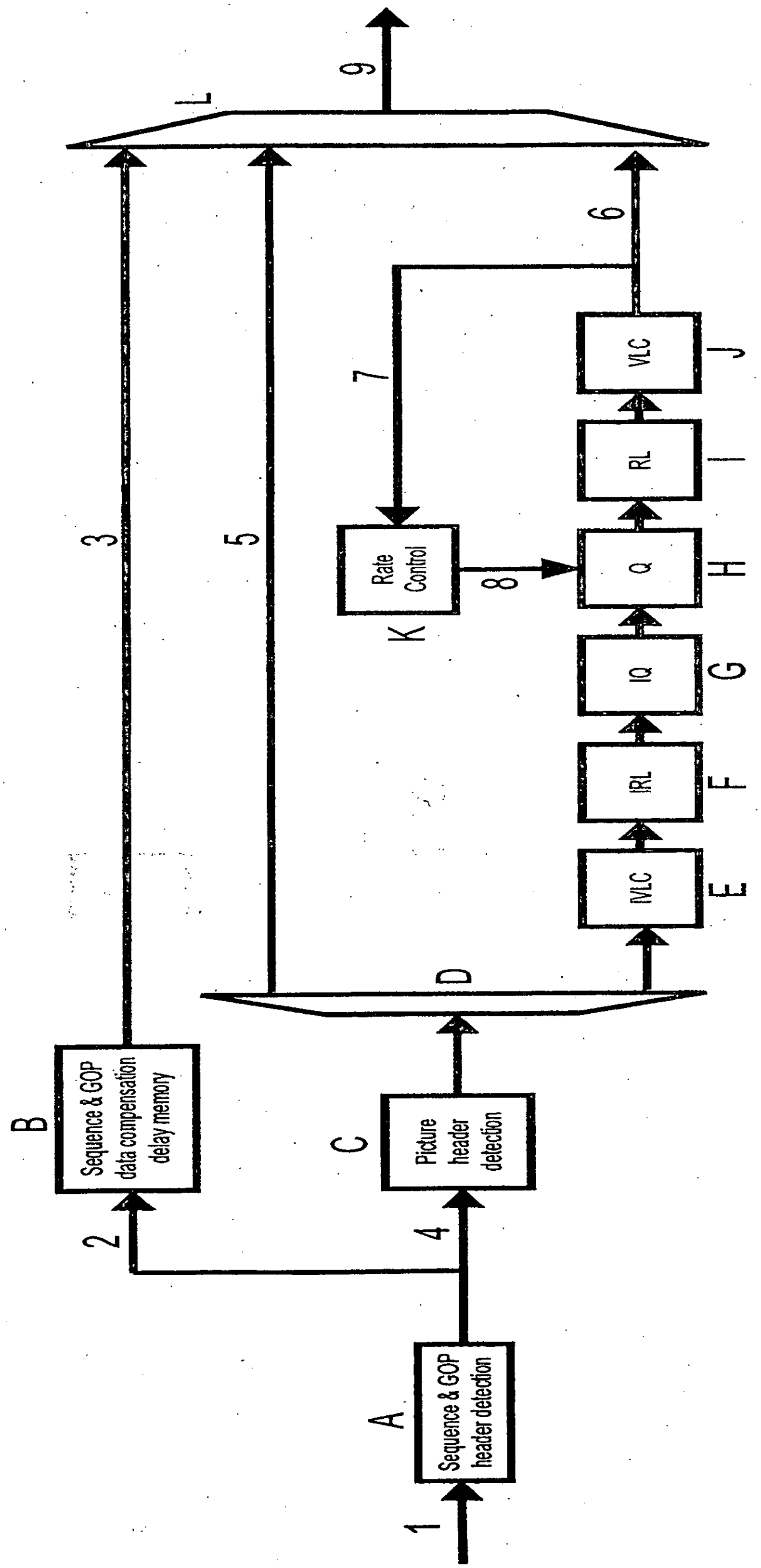
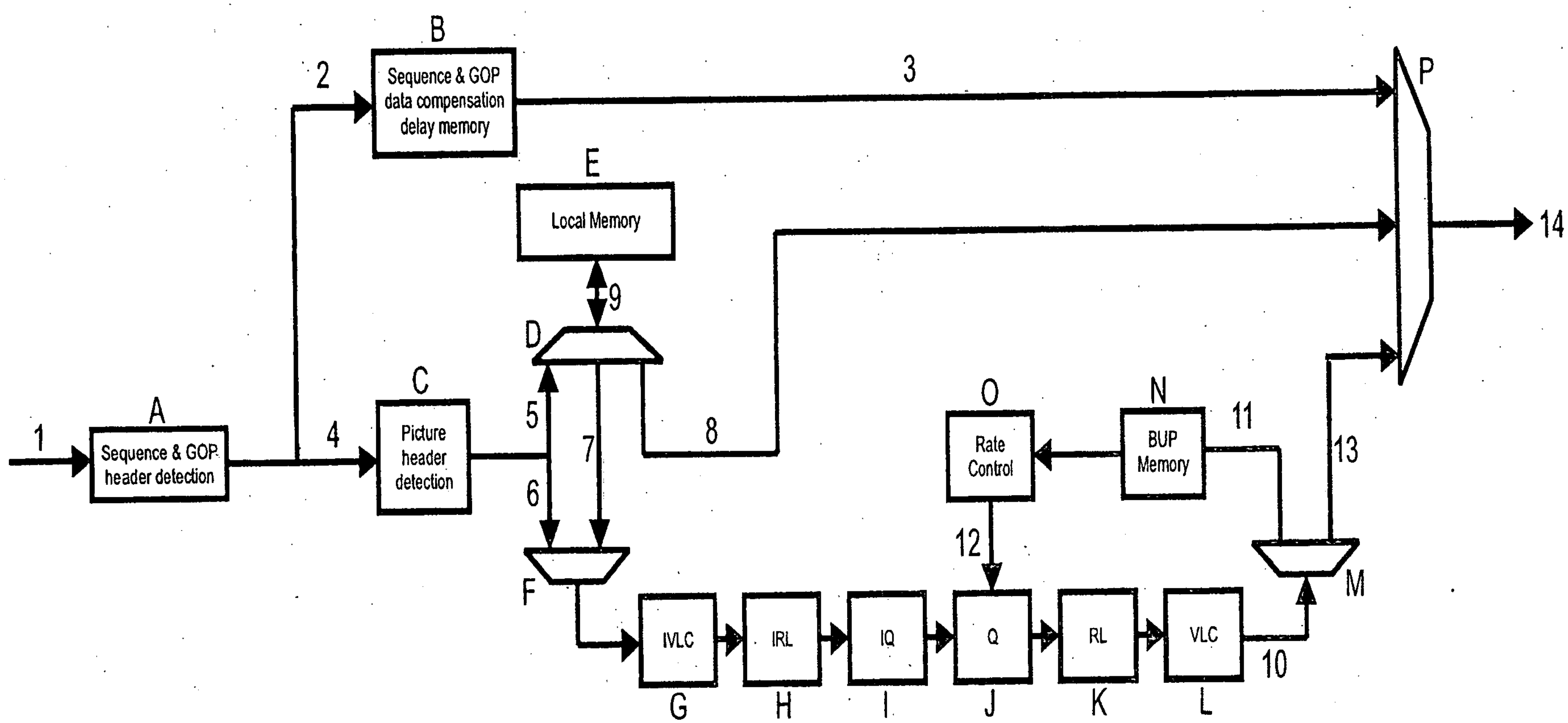

quantization parameter
input bitstream dct section
IVLC
IRL
IQ
Q
RL
VLC
output bitstream dct section

FIG. 4 bis



1
A
Sequence & GOP header detection
2
4
B
Sequence & GOP data compensation delay memory
C
Picture header detection
3
5
D
IVLC
E
IRL
F
IQ
G
Q
H
RL
I
VLC
J
Rate Control
K
7
8
6
L
9

FIG. 5



1
A
Sequence & GOP header detection
2
B
Sequence & GOP data compensation delay memory
3
4
C
Picture header detection
E
Local Memory
9
D
5
6
7
8
F
IVLC
IRL
IQ
Q
RL
VLC
G
H
I
J
K
L
O
Rate Control
N
BUP Memory
11
12
13
10
M
P
14

FIG. 6